REMARKS

The Examiner has objected to the Specification as failing to provide proper antecedent basis for the claimed subject matter. Specifically, the Examiner has argued that "the specification lacks antecedent basis for the newly recited claim 56." Applicant respectfully asserts that such objection has been avoided in view of the clarification made to the claim, as presently incorporated into each of the independent claims.

The Examiner has rejected Claim 56 under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. Additionally, the Examiner has rejected Claim 56 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Applicant respectfully asserts that such rejections have been overcome by virtue of the clarifications made hereinabove to the subject matter of such claim, as presently incorporated into each of the independent claims.

The Examiner has rejected Claims 1-3, 5-10, 13-14, 16-18, 20-25, 28-29, 31, and 55 under 35 U.S.C. §103(a) as being unpatentable over Trcka et al. (U.S. Patent No. 6,453,345), in view of Stevens (TCP/IP Illustrated). In addition, the Examiner has rejected Claims 4, 19, 32-38, 40-47, and 49-52 under 35 U.S.C.§103(a) as being unpatentable over Trcka, in view of Stevens, and further in view of Cheriton (U.S. Patent No. 7,054,930). Applicant respectfully disagrees with such rejections, especially in view of the amendments made hereinabove to the independent claims. Specifically, applicant has amended the independent claims to at least substantially include the subject matter of former dependent Claims 47 and 56.

With respect to independent Claims 1 and 16, the Examiner has relied on Pages 6-11 of the Stevens reference to make a prior art showing of applicant's claimed "reassembling one or more of the incoming datagrams into a segment structured in compliance with a transport protocol layer" (see the same or similar, but not necessarily identical language in the aforementioned independent claims).

Specifically, the Examiner has argued that "[i]t was well known that in the Internet Protocol there are multiple layers and that each layer contains different modules, such as the TCP module and the UDP module of the transport layer" and that "[i]t was also well known that in order to get to the data in the application layer packet, such as the payload and the packet type, the transport layer module must process the transport layer packet to reveal the application layer packet," as "evidenced by Stevens Pages 6-11."

Applicant respectfully disagrees. First, applicant respectfully asserts that the excerpt from the Stevens reference relied on by the Examiner merely relates to TCP/IP layering (see Page 6) and states that "[t]here are more protocols in the TCP/IP protocol suite" (see Page 6) "at different layers in the TCP/IP protocol suite" (see Figure 1.4 caption on Page 6). Clearly, only disclosing that multiple protocols exist at different layers in the TCP/IP protocol suite, as in the Stevens excerpt, fails to specifically teach "**reassembling** one or more of the incoming datagrams into a segment structured in compliance with a transport protocol layer" (emphasis added), as claimed.

Second, it also seems that the Examiner has relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has stated that "[i]t was also well known that in order to get to the data in the application layer packet, such as the payload and the packet type, the transport layer module must process the transport layer packet to reveal the application layer packet," as noted above. Applicant respectfully asserts that simply arguing that it was well known to process a transport layer packet to reveal an application layer packet, as noted by the Examiner, fails to even teach or suggest "**reassembling** one or more of the **incoming datagrams** into a segment structured in compliance with a transport protocol layer" (emphasis added), as claimed.

Thus, in response to the Examiner's apparent reliance on Official Notice in rejecting applicant's specific claim language, applicant again points out the remarks above that clearly show the manner in which some of such claims further distinguish

Trcka. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP below.

"If the applicant traverses such an [Official Notice] assertion the examiner should cite a reference in support of his or her position." See MPEP 2144.03.

In the Office Action dated 02/19/2008, the Examiner has cited Col. 14, lines 62-67 of Trcka and has argued that "the fact that Trcka disclosed performing scanning on upper layer files which rely upon TCP/IP for transmission, renders obvious the use of the TCP/IP stack of protocols as well as the OSI protocol model." In addition, the Examiner has argued that "as evidenced by Stevens on Page 148 Section 11.5, as well as Pages 6-11, TCP/IP performs 'reassembling one or more of the incoming datagrams into a segment structured in compliance with a transport protocol layer' when receiving packet data and converting the packet data to upper protocol (application layer)."

Applicant respectfully disagrees and notes that the above excerpt from Trcka relied on by the Examiner merely discloses that "virus checking can be performed on all incoming FTP (File Transfer Protocol) and HTTP files from unknown sites" (Col. 14, lines 62-64). Additionally, the excerpts from Stevens disclose that "[w]hen an IP datagram is fragmented, it is not reassembled until it reaches its final destination" and that "[t]he IP layer at the destination performs the reassembly" (see Section 11.5, paragraph 2). Further, the excerpts from Stevens disclose that "[t]here are more protocols in the TCP/IP protocol suite" (see Page 6) "at different layers in the TCP/IP protocol suite" (see Figure 1.4 caption on Page 6).

However, merely disclosing that virus checking can be performed on incoming FTP and HTTP files from unknown sites, as in Trcka, in addition to generally disclosing that when an IP datagram is fragmented it is not reassembled until it reaches its final destination, and that multiple protocols exist at different layers in the TCP/IP protocol suite, as in Stevens, fails to teach "<u>reassembling</u> one or more of the incoming datagrams into a segment structured in compliance with a transport protocol layer" (emphasis added), in the context specifically claimed by applicant.

Additionally, it seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has "relied upon... the fact that segmentation and reassembly of datagrams is well known in TCP/IP, which is, and was at the time of invention, a very well known and used standard." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claims are distinguished from Trcka and Stevens. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

With respect to independent Claim 1, the Examiner has relied on column 13, lines 32-49 from the Trcka reference to make a prior art showing of applicant's claimed "protocol-specific module processing each reassembled datagram based on the transport protocol layer employed by the reassembled datagram."

Applicant respectfully asserts that the excerpt from the Trcka excerpt relied on by the Examiner merely discloses that a "Post-Capture Processing Module 98 processes the packets based on protocol-specific packet fields." Clearly, only generally disclosing a module that processes packets based on protocol specific packet fields, as in Trcka, fails to meet applicant's claimed "protocol-specific module processing each reassembled datagram based on the transport protocol layer employed by the reassembled datagram" (emphasis added), as claimed.

In the Office Action dated 02/19/2008, the Examiner has argued that Trcka...disclose[s] in col. 14 lines 49-65, that the Post-Capture Processing Module automatically reads in and analyzes the data from the recorders which store continuous packet data." The Examiner has further argued that "Trcka did disclose in col. 14, lines 49-65, that the Post-Capture Processing Module automatically reads in and analyzes the data from the recorders which store continuous packet data" and that "Trcka further disclosed that the data was analyzed based upon the specific file types, which is application layer data." Additionally, the Examiner has argued that "[a]s evidenced by Stevens... the TCP/IP protocol stack performs processing of datagrams based on the transport protocol layer employed by the reassembled datagram in order to produce application layer data from packet data."

Applicant respectfully disagrees and notes that the above reference excerpt from Trcka relied on by the Examiner merely disclose that "the Good-Data Cyclic Recorder 82 and the Bad-Data Cyclic Recorder 84 provide a temporary record of the good and bad packet data (respectively)," that "the cyclic recorders 82, 84 can typically store about 24 hours of continuous traffic data," and that "[w]hen automated monitoring is enabled, the software routines of the Post-Capture Processing Module 98 automatically read-in and analyze the data from the Good-Data Cyclic Recorder 82 and/or the Bad-Data Cyclic Recorder 84 as or shortly after it is recorded" (Trcka, Col. 14, lines 49-60). Additionally, the Stevens reference merely discloses that "IP must add some type of identifier to the IP header that it generates, to indicate the layer to which the data belongs" and that "IP handles this by storing an 8-bit value in its header called the *protocol* field" (Page 10, first paragraph), and further discloses that "[w]hen an IP datagram is fragmented, it is not reassembled until it reaches its final destination" and that "[t]he IP layer at the destination performs the reassembly" (see Section 11.5, paragraph 2).

However, merely disclosing that a Post-Capture Processing Module <u>reads in and analyzes</u> data from <u>a temporary record of good and bad packet data</u>, as in Trcka, in addition to disclosing that IP indicates the layer to which data belongs in a protocol field, and that the IP layer at a destination performs the reassembly of a <u>fragmented IP datagram</u>, as in Stevens, fails to teach a "<u>protocol-specific module</u> <u>processing</u> each <u>reassembled datagram</u> based on the transport protocol layer employed by the reassembled datagram" (emphasis added), in the context claimed by applicant.

Additionally, it again seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has argued that it is "well known in the art [that] the TCP/IP protocol stack performs processing of datagrams based on the transport protocol layer employed by the reassembled datagram in order to produce application layer data from packet data." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claims are distinguished from Trcka and Stevens. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

With respect to independent Claims 32 and 41, the Examiner has relied on Col. 2, lines 29-34; Col. 4, lines 2-11; Col. 7, lines 28-32; and Col. 12, lines 29-40 of the Trcka reference to make a prior art showing of applicant's claimed "receiving copies of datagrams transiting a boundary of a network domain into an incoming packet queue" (see the same or similar, but not necessarily identical language in the aforementioned independent claims).

Applicant respectfully asserts that the excerpts from the Trcka reference relied on by the Examiner simply teach that "the system captures and records the packets passively" (Col. 2, lines 30-31), "software which continuously routes at least some of the passively-captured traffic data to a cyclic data recorder" (Col. 4, lines 4-5), "[t]he archival recording generated by the above-described process is in essence a complete replica of all valid network traffic" (Col. 7, lines 28-30), and "the archival data stream generated by the Archival Data Processing Module 90...is...routed to...enable the automated analysis of such data" (Col. 12, lines 29-33).

Only generally disclosing capturing and recording packets, as in the Trcka excerpts, does not teach "receiving <u>copies</u> of datagrams transiting a boundary of a network domain" (emphasis added), as claimed. In fact, the Trcka reference expressly discloses that the <u>archival recording</u> is in essence a complete replica of all valid network traffic, as noted above, which does not meet applicant's specifically claimed "receiving copies of datagrams transiting a boundary of a network domain" (emphasis added), as claimed by applicant.

In the Office Action dated 02/19/2008, the Examiner has argued that "Col. 19 Paragraph 2 and Fig. 8 [of Trcka] disclose a Firewall, which is a network boundary, wherein the data packets are captured from both sides of the firewall" and that "Trcka teaches that the packet data is a passively captured replica of the network traffic, and a replica is a copy." Further, the Examiner has argued that "Trcka states that the passively generated data stream 'represents the traffic present on the network' (See Trcka Col. 10 Lines 59-63, which further implies that the passively generated data stream is not the traffic present on the network, but rather it is a copy of the traffic present on the network." In addition, the Examiner has argued that "'the traffic present on the network' falls within the scope of a 'packet stream.'"

Applicant respectfully disagrees and asserts that the excerpts relied on by the Examiner merely disclose a "configuration in which a single system 60' is used to monitor both pre-firewall and post firewall traffic," where "one [Archival Data Processing Module] 90A processes traffic on the Internet side of a conventional firewall 150, while the other ADPM 90B processes traffic on the internal-network side of the firewall 150" and where "[e]ach ADPM 90A, 90B passively receives traffic data from a different respective network interface card (not shown), and forwards the processed raw traffic data to a respective Archival Media Unit/Good-data Cyclic Recorder pair" (Col. 19, lines 9-20 - emphasis added). Additionally, the excerpts disclose "traffic capture components which run continuously in the background to passively generate a data stream that represents the traffic present on the network 30" where "the functions performed by these components include filtering and encrypting the incoming packet stream, and inserting date/time stamps into the packet stream" (Col. 10, lines 59-66 - emphasis added).

However, merely disclosing that separate Archival Data Processing Modules process traffic on either side of a firewall, where the ADPMs receive traffic from different network cards, in addition to disclosing that traffic capture components passively generate a data stream and perform filtering, encryption, and time stamping on the packet stream, as in Trcka, fails to disclose "receiving copies of datagrams transiting a boundary of a network domain," much less "receiving copies of datagrams transiting a boundary of a network domain into an incoming packet queue" (emphasis added), as claimed by applicant.

Further, with respect to independent Claims 32 and 41, the Examiner has relied on Pages 6-11 of the Stevens reference in addition to the rejection of Claim 1 to make a prior art showing of applicant's claimed "reassembling one or more such datagrams from the incoming packet queue into network protocol packets, each staged in a reassembled packet queue" (see the same or similar, but not necessarily identical language in the aforementioned independent claims).

First, applicant respectfully asserts that the excerpt from Stevens relied on by the Examiner merely relates to TCP/IP layering (see Page 6) and states that "[t]here are more protocols in the TCP/IP protocol suite" (see Page 6) "at different layers in the TCP/IP protocol suite" (see Figure 1.4 caption on Page 6). Clearly, only disclosing that multiple protocols exist at different layers in the TCP/IP protocol suite, as in Stevens, fails to specifically teach "**reassembling** one or more such datagrams," and particularly not "reassembling one or more such datagrams from the incoming packet queue into network protocol packets, each staged in a reassembled packet queue" (emphasis added), as claimed. Applicant emphasizes that Pages 4-11 in Stevens, as relied on by the Examiner, does not even suggest any sort of reassembling, incoming packet queue or reassembled packet queue, and especially not in the manner claimed by applicant.

Second, it also seems that the Examiner has relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has stated in the rejection of Claim 1 relied on by the Examiner that "[i]t was also well known that in order to get to the data in the application layer packet, such as the payload and the packet type, the transport layer module must process the transport layer packet to reveal the application layer packet," as noted above. Applicant respectfully asserts that simply arguing that it was well known to process a transport layer packet to reveal an application layer packet, as noted by the Examiner, fails to even suggest "**reassembling** one or more such datagrams from the incoming packet queue into network protocol packets, each staged in a reassembled packet queue" (emphasis added), as claimed.

Thus, in response to the Examiner's apparent reliance on Official Notice in rejecting applicant's specific claim language, applicant again points out the remarks above that clearly show the manner in which some of such claims further distinguish Trcka. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. See MPEP 2144.03 above.

In the Office Action dated 02/19/2008, the Examiner has argued that "Trcka teaches that the cyclic data recorder temporarily stores the passively captured traffic data (packet stream), which meets the limitation of 'the incoming packet queue'" and that "Trcka further teaches the data is read out of the cyclic data recorder to check for viruses, as can be seen in Col. 4 Paragraph 1." Furthermore, the Examiner has argued that "Trcka further disclosed that the scanning is performed on files such as HTTP files, FTP files, etc. (See Col. 14 Lines 61-64)" and that "[a]s evidenced by the teachings of Stevens, converting the packets to HTTP files or FTP files involves demultiplexing from the network layer, which includes IP, to the transport layer, which includes TCP, to the application layer, which includes HTTP and FTP." In addition, the Examiner has argued that "Stevens further evidences that the demultiplexing from IP to TCP involves reassembly of datagram fragments, as seen on Page 148 of Stevens."

Applicant respectfully disagrees and asserts that the excerpts from Trcka relied on by the Examiner merely disclose "software which continuously routes at least some of the passively-captured traffic data to a cyclic data recorder," where "[t]he cyclic data recorder… is used to temporarily store the traffic data for automated post-capture analysis" and where "a real-time monitoring application reads the traffic data from the cyclic recorder on a first-in-first-out basis and checks for pre-programmed anomalies" (Col. 4, lines 3-11). Further, the excerpts teach that "[b]ecause the traffic data is analyzed only after being passively captured, thorough analyses (extensive virus checks, reconstruction of transaction sequences, etc.) can be performed without any interruption to the normal flow of data on the network" (Col. 4, lines 13-17).

Further, the excerpts from Trcka disclose that "[a]ny of a variety of known security checks can be performed on the packet data [by the Post-Capture Processing Module] at this stage," for example, "virus checking can be performed on all incoming FTP (File Transfer Protocol) and HTTP files from unknown sites" (Col. 14, lines 61-64). Further still, the excerpt from Stevens discloses that that "[w]hen an IP datagram is fragmented, it is not reassembled until it reaches its final destination" and that "[t]he IP layer at the destination performs the reassembly" (see Page 148, Section 11.5, paragraph 2).

However, merely disclosing a cyclic data recorder which temporarily stores traffic data, where data is read from the cyclic recorder on a first-in-first-out basis and checked for pre-programmed anomalies, in addition to generally disclosing that thorough analyses such as virus checks can be performed without data flow interruption due to the passive capture of data, and that virus checking can be performed on all incoming FTP and HTTP files from unknown sites, as in Trcka, in addition to disclosing that an IP layer at the destination performs the reassembly of a fragmented IP datagram, as in Stevens, fails to disclose "reassembling one or more such datagrams from the incoming packet queue into network protocol packets, each staged in a reassembled packet queue" (emphasis added), as claimed by applicant.

Additionally, it again seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has argued that "[i]t is obvious that the after defragmenting at the

IP layer, and prior to demultiplexing from the defragmented IP datagram to the TCP Segment, the defragmented IP datagram would have to be stored somewhere, or it would be lost" and that "this storage is a queue, [which] meets the limitations of the claim language." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claims are distinguished from Trcka and Stevens. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

Moreover, with respect to independent Claims 32 and 41, the Examiner has relied on Col. 3, line 66-Col. 4, line 16 in Trcka to make a prior art showing of applicant's claimed "scanning each network protocol packet from the reassembled packet queue to ascertain an infection of at least one of a computer virus and malware" (see the same or similar, but not necessarily identical language in the aforementioned independent claims).

Applicant respectfully asserts that such excerpt from Trcka only discloses that "a real-time monitoring application reads the traffic data from the cyclic recorder on a first-in-first-out basis and checks for pre-programmed anomalies." However, applicant notes that Trcka only discloses "software which continuously routes at least some of the passively-captured traffic data to a cyclic data recorder" (Col. 4, lines 3-5).

Thus, the excerpt from Trcka relied on by the Examiner only discloses reading traffic data from a cyclic recorder on a first-in-first-out basis and checking such traffic data for pre-programmed anomalies, <u>where the traffic data read from the cyclic recorder includes passively-captured traffic data</u>. To this end, applicant respectfully points out that checking passively-captured traffic data stored in a cyclic recorder, as in Trcka, fails to specifically meet applicant's claimed "scanning each network protocol packet <u>from the</u> **<u>reassembled</u>** <u>packet queue</u> to ascertain an infection of at least one of a computer virus and malware" (emphasis added), particularly where "reassembl[ed]...datagrams [are] each staged in [the] reassembled packet queue," in the context claimed by applicant.

In the Office Action dated 02/19/2008, it seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has argued that "this limitation has been shown as obvious in view of Trcka as evidenced by Stevens" and that "[i]n this combination, because the reassembly occurs prior to the packets becoming files, as is evidenced by Stevens, and because the files, which are demultiplexed from the packets, are what is being scanned, it is obvious that each reassembled packet is scanned." Additionally, the Examiner has further argued that "as discussed above, each reassembled packet is obviously stored in 'a queue'." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claims are distinguished from Trcka and Stevens. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

With respect to independent Claim 32, the Examiner has relied on Page 11 in Stevens and the rejection of Claim 1 to make a prior art showing of applicant's claimed technique "wherein a protocol-specific module processes each reassembled datagram based on an upper protocol layer employed by the reassembled datagram."

Applicant respectfully asserts that Page 11 in Stevens only discloses demultiplexing in which "an Ethernet frame is received at the destination host [and] starts its way up the protocol stack [where] all the headers are removed by the appropriate protocol box." Clearly, such disclosure of demultiplexing does not even suggest a "protocol-specific module [that] processes **each reassembled datagram** based on an upper protocol layer employed by the reassembled datagram" (emphasis added), as claimed.

In addition, as noted above with respect to the rejection of Claim 1, as relied on by the Examiner, Col. 13, lines 32-49 in Trcka merely discloses that a "Post-Capture Processing Module 98 processes the packets based on protocol-specific packet fields." Clearly, only generally disclosing a module that processes packets based on protocol specific packet fields, as in Trcka, fails to meet applicant's claimed "protocol-specific module [that] processes each reassembled datagram based on an upper protocol layer employed by the reassembled datagram" (emphasis added), as claimed.

In the Office Action dated 02/19/2008, the Examiner has argued that "Trcka disclosed creating the application layer files for scanning, such as HTTP files, and FTP files, and as discussed above, TCP/IP reassembles fragmented datagrams at the IP layer, then sends the IP datagram to the transport layer protocol corresponding to that datagram, which demultiplexes the datagram based upon the protocol for that packet, such as TCP or UDP."

Applicant respectfully disagrees and again notes that Trcka merely discloses that "virus checking can be performed on all incoming FTP (File Transfer Protocol) and HTTP files from unknown sites" (Col. 14, lines 62-64). Additionally, Stevens discloses that "[w]hen an IP datagram is fragmented, it is not reassembled until it reaches its final destination" and that "[t]he IP layer at the destination performs the reassembly" (see Section 11.5, paragraph 2). Further, Stevens discloses that "[t]here are more protocols in the TCP/IP protocol suite" (see Page 6) "at different layers in the TCP/IP protocol suite" (see Figure 1.4 caption on Page 6).

However, merely disclosing that virus checking can be performed on incoming FTP and HTTP files from unknown sites, as in Trcka, in addition to generally disclosing that when an IP datagram is fragmented it is not reassembled until it reaches its final destination, and that multiple protocols exist at different layers in the TCP/IP protocol suite, as in Stevens, fails to teach a technique "wherein a protocol-specific module processes each reassembled datagram based on an upper protocol layer employed by the reassembled datagram" (emphasis added), as specifically claimed by applicant.

Additionally, it seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has argued that "[applicant's claimed language] is simply another obvious feature of TCP/IP, as evidenced by Stevens." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claims are distinguished from Trcka and Stevens. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *In re Vaeck,*947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Applicant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest <u>all</u> of the claim limitations, as noted above. Nevertheless, despite such paramount deficiencies and in the spirit of expediting the prosecution of the present application, applicant has at least substantially incorporated the subject matter of former dependent Claims 47 et al. and 56 into the independent claims.

With respect to the subject matter of former Claim 47 (now at least substantially incorporated into the independent claims), the Examiner has relied on Col. 2, lines 16-24, Col. 3, lines 29-45, and Claim 7 from the Cheriton reference to make a prior art showing of applicant's claimed "sending a spoofed network protocol packet responsive to an occurrence of at least one of the infection and the network attack" (see this or similar, but not necessarily identical language in the independent claims).

Applicant respectfully notes that the excerpts from Cheriton relied on by the Examiner merely disclose "filter[ing] harmful data" where "a netflow directory and flow analyzer are used to detect harmful network flows...which needs to be filtered" (Col. 3, lines 34-42), and "generating a filter to prevent packets corresponding to said detected potentially harmful network flows from passing through said second network device" (Claim 7).

However, merely filtering packets corresponding to harmful network flows, as in Cheriton, fails to teach "sending a spoofed network protocol packet responsive to an occurrence of at least one of the infection and the network attack" (emphasis added), as claimed by applicant.

Additionally, with respect to the subject matter of former Claim 56 (now at least substantially incorporated into the independent claims), the Examiner has failed to provide a specific prior art rejection of such subject matter under 35 U.S.C. 103. Thus, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Applicant further notes that the prior art is also deficient with respect to the dependent claims. For example, with respect to dependent Claim 4 et al., the Examiner has relied on Col. 2, lines 16-24; Col. 3, lines 29-45; and Claim 7 in Cheriton to make a prior art showing of applicant's claimed technique "wherein the antivirus scanner terminates the transient packet stream if the reassembled segment is not infected with at least one of a computer virus and malware."

Specifically, the Examiner has argued that Cheriton teaches "generation and refinement of filters for stopping the attack packets, and forwarding these filters upstream." Applicant respectfully disagrees and asserts that stopping attack packets does not meet, and even *teaches away* from applicant's claimed technique "wherein the antivirus scanner terminates the transient packet stream if the reassembled segment is **not** infected with at least one of a computer virus and malware" (emphasis added), as claimed.

In addition, applicant notes that the excerpts from Cheriton relied on by the Examiner merely disclose "filter[ing] harmful data" where "a netflow directory and flow analyzer are used to detect harmful network flows...which needs to be filtered" (Col. 3, lines 34-42), and "generating a filter to prevent packets corresponding to said detected potentially harmful network flows from passing through said second network device" (Claim 7). Thus, Cheriton clearly discloses filtering harmful data, and not a technique "wherein the antivirus scanner terminates the transient packet stream if the reassembled segment is **not** infected with at least one of a computer virus and malware" (emphasis added), as claimed.

In the Office Action dated 02/19/2008, the Examiner has argued that "Denial of Service attack packets are not infected with viruses or malware, but rather... either contain invalid parameters or are transmitted in large quantities." Additionally, the examiner has argued that "only one of the possibilities has been addressed by the claim language, and says nothing about the situation when the packet is infected."

Applicant respectfully disagrees and again notes that the excerpts relied on by the Examiner merely teach filtering packets corresponding to harmful network flows, which does not disclose a technique "wherein the antivirus scanner terminates the transient packet stream if the reassembled segment is not infected with at least one of a computer virus and malware" (emphasis added), as claimed by applicant.

Additionally, it seems that the Examiner has again relied on an Official Notice argument to reject applicant's specific claim language. For example, applicant notes that the Examiner has argued that "the teachings of Cheriton do render obvious the scenario when the packets are not infected and the stream is stopped." Applicant respectfully disagrees and again points out the remarks above that clearly show the manner in which applicant's aforementioned claim is distinguished from Cheriton. Applicant thus formally requests a specific showing of the subject matter in ALL of the claims in any future action. Note excerpt from MPEP above.

With respect to dependent Claim 55, the Examiner has relied on Col. 14, lines 61-67 in Trcka to make a prior art showing of applicant's claimed technique "wherein the incoming datagrams include IP datagrams that are reassembled into TCP segments."

Applicant respectfully asserts that the excerpt from Trcka relied on by the Examiner only teaches that "[a]ny of a variety of known security checks can be performed on the packet data at this stage," such as performing "virus checking... on all incoming FTP (File Transfer Protocol) and HTTP files from unknown sites." Clearly, only disclosing performing security checks on packet data, as in Trcka, fails to even suggest that "incoming datagrams include IP datagrams that are reassembled into TCP segments," as applicant claims.

In the Office Action dated 02/19/2008, the Examiner has argued that "the applicants have misconstrued Trcka by stating that the security checks are performed on packet data" and that "Trcka clearly disclosed performing security checks on files, which are created from packet data (see Trcka Col. 14 Lines 49-64)."

Applicant respectfully disagrees and notes that the excerpt relied on by the Examiner merely discloses that "the Good-Data Cyclic Recorder 82 and the Bad-Data Cyclic Recorder 84 provide a temporary record of the good and bad packet data," where "the software routines of the Post-Capture Processing Module 98 automatically read-in and analyze the data from the Good-Data Cyclic Recorder 82 and/or the Bad-Data Cyclic Recorder 84 as or shortly after it is recorded" and that where "[a]ny of a variety of known security checks can be performed on the packet data at this stage" (Col. 14, lines 49-62 - emphasis added). Additionally, the excerpt teaches that "virus checking can be performed on all incoming FTP (File Transfer Protocol) and HTTP files from unknown sites" (Col. 14, lines 62-64).

However, merely disclosing that cyclic recorders provide a temporary record of packet data, which is read and analyzed by software routines that perform security checks on the packet data, and that virus checking may be performed on incoming FTP and HTTP files from unknown sites, as in Trcka, fails to even suggest a technique "wherein the incoming datagrams include IP datagrams that are reassembled into TCP segments" (emphasis added), as claimed by applicant.

Again, since at least the third element of the *prima facie* case of obviousness has not been met, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Still yet, applicant brings to the Examiner's attention the subject matter of new Claim 58 below, which is added for full consideration:

> "wherein the packet receiver maintains each protocol-specific queue at a constant size in accordance with the antivirus scanner" (see Claim 58).

Again, a notice of allowance or a proper prior art showing of all of applicant's claim limitations, in combination with the remaining claim elements, is respectfully requested.

To this end, all of the independent claims are deemed allowable. Moreover, the remaining dependent claims are further deemed allowable, in view of their dependence on such independent claims.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 50-1351 (Order No. NAI1P393/01.162.01).

Respectfully submitted,
Zilka-Kotab, PC.

/KEVINZILKA/

Kevin J. Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
408-505-5100